

LUXOR

Datorer

Options-PROM för: ABC800	ABC7-21, art nr: 64 90140-06
ABC802	Optionsdelen av DOS-Options-PROM ABC32-11, art nr: 64 90190-02
ABC806	ABC76-1, art nr: 64 90238-01

1. ALLMÄNT

Options-PROM enl ovan innehåller drivrutiner för kommunikation via I/O kort eller via datorns två serieutgångar för:

- o Olika skrivare.
- o Asynkron terminal.
- o Datavisionsterminal (ej ABC800M, ABC806 endast tillsammans med drivrutin laddad från skiva).

På kanal B (V24:) jobbar man med en sändnings- och mottagningsbuffert på 80 tecken, som är interruptstyrd. Vidare finns det rutiner för kontroll av paritet och XON/XOFF. När man gör OPEN eller PREPARE kan man kontrollera att skrivaren är ansluten, detta för att man inte skall bli hängande vid PRINT #nr. Erhålls inte DCD ges felmeddelande 42 (enheten ej klar).

Följande kommandon och instruktioner är tillgängliga:

```
LIST PR:parameterblock
LOAD PR:parameterblock
PREPARE "PR:parameterblock" AS FILE nr
OPEN "PR:parameterblock" AS FILE nr
LIST V24:parameterblock
LOAD V24:parameterblock
PREPARE "V24:parameterblock" AS FILE nr
OPEN "V24:parameterblock" AS FILE nr
```

Vid initiering med PREPARE ges formfeed (FF), dvs ny sida, vid stängning (CLOSE) av filen.

Parameterblocket består av upp till 11 tecken förutom enhetsnamn. För information om parameterblocket - se avsnitt 3. Kanal A (märkt CH. A) öppnas med PR: och kanal B (märkt CH. B) öppnas med V24:.

2. "SPLIT SPEED"

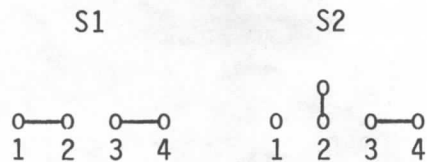
Datorn är vid leverans byglad för samma mottagnings- och sändningshastighet, men kan byglas om till split speed, dvs olika hastigheter på sändning och mottagning.

Betr. ABC802 och ABC806, se avsnitt om BYGLINGAR i bruksanvisningen.

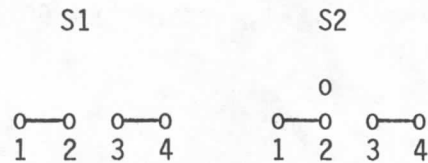
I ABC800 finns ett byglingsfält, S1 och S2, på PU-kortet, till höger om anslutningskabeln till tangentbordet.

ABC800®

I ABC800 ser byglingsfältet ut så här vid leverans:



För att erhålla split speed ska S2 byglas om enligt nedan:



3. PARAMETERBLOCKETS UPPBYGGNAD

Om man inte vill använda de standardparametrar (default) som finns definierade i options-PROMmet, kan man byta ut en eller flera parametrar, utan att skriva om hela parameterblocket. Vill man ändra någon parameter i parameterblocket utan att skriva om samtliga parameter skriver man bara den nya parametern. I övriga positioner skrivs '0'. Exempel:

OPEN 'PR:00002000.00B' AS FILE Nr

Standardvärdena är markerade med def (default) nedan. Angivna standardvärden gäller för både PR: och V24:.

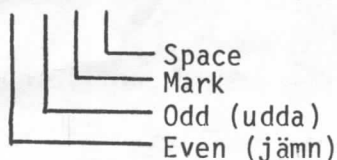
1:a tecknet anger terminal eller printertyp.

Bokstav	Typ	
C	Centronics	(endast PR:)
P	SP1	(endast PR:)
T	Terminal	(endast V24: - CH. B)
U	UART	(endast PR:)
V (def)	V24 simulerad UART	(PR: - CH. A eller V24: - CH.B)
W	Datavision	(endast V24: - CH. B)

Exempel: LIST PR V

2:a tecknet anger paritet och XON/XOFF.

A B C D - Ingen paritetscheck+XON/XOFF
 F G H I - Paritetscheck
 E O M S - Ingen paritetscheck
 T U V W - Paritetscheck + XON/XOFF



Som standardvärde gäller bokstaven S (paritet space och ingen paritetscheck).

Exempel: LIST PR:VS

3:e tecknet anger antalet "NUL" (ASCII-tecken nr noll) som sänds efter radframmatning.

Bokstav	Antal
A (def)	0
B	2
C	4
.	.
Z	50

Exempel: LIST PR:VSA

4:e tecknet anger antal tecken/rad.

Siffra	Antal
1	40
2	72
3 (def)	80
4	120
5	132
6	158
7	254

Exempel: LIST PR:VSA3

För att detta ska ha någon inverkan måste formatstyrtecken som genererar CRLF väljas (se 6:e tecknet).

5:e tecknet anger antal rader som ska hoppas över vid sidslut (perforeringsskip).

Siffra	Antal
0	0
1	1
2	2
3	3
4	4
5	5
6 (def)	6
7	7
8	8
9	9

Exempel: LIST PR VSA36

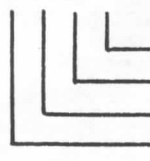
6:e tecknet anger formatstyrtecken för skrivare samt överförings-
typ i terminalmode.

A E I M - Ej CRLF vid radslut, ej FF-simulering

B F J N - Ej CRLF vid radslut, FF-simulering

C G K O - CRLF vid radslut, ej FF-simulering

D H L P - CRLF vid radslut, FF-simulering

	Halv duplex, om CR mottages genereras LF
	Full duplex, om CR mottages genereras LF
	Halv duplex, ej extra LF om CR mottages
	Full duplex, ej extra LF om CR mottages

CRLF = Vagnretur och radframmatning.

CR = Vagnretur (Carriage return).

LF = Radframmatning (Line feed).

FF = Ny sida (Form feed).

Som standardvärde gäller B (Ej CRLF vid radslut, FF-simulering för skrivare och full duplex, ej extra LF om CR mottages i terminalmode).

FF-simulering fungerar endast mot skrivare. FF-simulering betyder att datorns printerrutin skickar ett antal LF (Line feed) för att komma till nästa sida. Ej FF-simulering betyder att printerrutinen skickar FF (Form Feed) som sedan tas om hand i skrivaren som gör sidframmatning. CR och LF generering bryter raden vid radslut efter angiven radlängd.

Exempel: LIST PR VSA36B

7:e och 8:e tecknen anger antal rader/sida.

Här anges antal rader per sida. Man kan ange högst 99 rader. Som standardvärde gäller 72 rader/sida, detta motsvarar en pappershöjd på 12".

OBS! Antalet måste alltid anges med två siffror.

Siffra	Rader/sida
--------	------------

01	1
02	2
.	.
.	.
72 (def)	72
.	.
.	.
99	99

Exempel: LIST PR:VSA36B72

9:e tecknet anger överföringshastighet (baudrate).

Om datorn byglas om för split speed anger denna parameter mottagningshastigheten. I annat fall gäller parametern för både sändnings- och mottagningshastighet.

OBS! Siffran skall föregås av en punkt.

Siffra	Hastighet (baud)
--------	------------------

0	75
1	110
2	300
3	600
4	1200
5 (def)	2400
6	4800
7	9600
8	19200

Exempel: LIST PR:VSA36B72.5

10:e tecknet anger sändningshastighet (gäller endast i split speed).

Siffra	Hastighet (baud)
--------	------------------

0	75
1	110
2	300
3	600
4	1200
5 (def)	2400
6	4800
7	9600
8	19200

Exempel: LIST PR VSA36B72.55

11:e tecknet.

Test sker genom att kontrollera DCD (pin 8 i kontakt CH. A och CH. B) vid OPEN eller PREPARE. Om DCD saknas får man felmeddelande 42, enheten ej klar.

Möjlighet att maska högsta biten i dataordet.

OBS! Om man är i mode E,F,G eller H bör man köra utan paritets test eller med paritet 'SPACE'.

Nedanstående gäller endast CH. B.

Normal XON/XOFF hantering innebär att options-PROMmet skickar XOFF när det är 50 tecken-positioner lediga i bufferten. XON skickas när det finns 70 eller fler tecken-positioner lediga i bufferten.

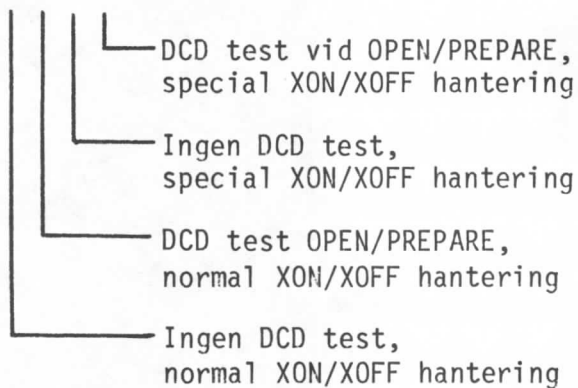
Special XON/XOFF hantering innebär att options-PROMmet skickar XON när man gör INPUT, INPUT LINE eller GET. XOFF sänds när man har fått alla data. Under tiden som man håller på att hämta data så fungerar XON/XOFF hanteringen som så kallad 'normal XON/XOFF hantering'.

A B C D - Bit 7 är oförändrad på inkommande data

E F G H - Bit 7 tas bort på inkommande data

I J K L - Reserverad

M N O P - Reserverad



Som default gäller bokstaven A.

Exempel: LIST PR:VSA36B72.55A

4. TERMINALMODE

Med options-PROM kan datorn användas som terminal på två olika sätt. Dels som bildskärmsterminal, se 4.1 nedan, eller tillsammans med BASICprogram, se 4.2.

4.1 Bildskärmsterminal

Vid användning av datorn som terminal motsvarar detta en ADM-3A bildskärmsterminal. När denna form av terminal används ska första tecknet efter V24: vara ett T. I övrigt byggs parameterblocket upp enligt ovan. Interfacet, som kommunikationen sker över, är RS232C/V24 anslutning. Dataordet består av 1 startbit, 7 databitar, 1 paritetsbit och 1 stoppbit. Övergång till terminalmode kan ske på något av följande sätt:

```
LOAD V24:TSA30B24.22A
```

```
OPEN "V24:TSA30B24.22A" AS FILE 1
INPUT #1,A
```

```
OPEN "V24:TSA30B24.22A" AS FILE 1
INPUT LINE #1,A
```

```
OPEN "V24:TSA30B24.22A" AS FILE 1
GET #1,A
```

Det är möjligt att fördefiniera PF-tangenter innan man går in i terminalmode. (PF-tangenterna för ABC802/806 med tangentbord ABC55 finns beskrivna i respektive bruksanvisning). När man trycker en definierad PF-tangent skickas den sträng som har definierats, t ex PF1 har definierats som "LOGIN" detta innebär att när PF1 trycks ner sänder options-PROMmet strängen "LOGIN". För att definiera en PF-tangent skapar man en strängvariabel, först i variabeln läggs ASCII-koden för funktionstangenten och därefter den sträng som skall skickas när funktionstangenten trycks ner. Sedan upprepas ASCII-kod och sträng för nästa funktionstangent etc. När man har definierat de funktionstangenter som man vill använda måste strängen avslutas med ASCIIkod 255. Se nedanstående exempel:

```
A=CHR(192)+"LOGIN"+CHR(13,193)+"HELP"+CHR(13,194)+"LIB"+CHR(13,255)
POKE PEEK2(65500)+8,VAROOT(A),SWAP%(VAROOT(A))
LOAD V24:TEA30E24.22A
```

Uthopp ur terminalmode kan ske på följande tre sätt:

1. Genom att trycka en PF-tangent (funktionstangent) som inte är fördefinierad. Detta ger felmeddelande 53. Om man använder ett BASICprogram kan detta fel hanteras med en felhanterare.
2. Genom att datorn tar emot ENQ (ASCII-värde 5). Detta ger felmeddelande 34. Om man använder ett BASICprogram kan detta fel hanteras med en felhanterare.
3. Genom att datorn tar emot STX (ASCII-värde 2). De efterföljande tecknen fram till CR (ASCII-värde 13) kommer att hamna i den strängvariabel som används. Detta gäller bara INPUT #nr och INPUT LINE #nr.

Följande tangentkoder används i terminalmode:

CTRL-S (ASCII-värde 19) stoppar utskriften på bildskärmen.

CTRL-Q (ASCII-värde 17) startar utskriften på bildskärmen.

<-- eller

CTRL-H (ASCII-värde 8) flyttar markören ett steg till vänster.

CTRL-J (ASCII-värde 10) flyttar ner markören en rad.

CTRL-K (ASCII-värde 11) flyttar upp markören en rad.

CTRL-L (ASCII-värde 12) flyttar markören ett steg till höger.

CTRL-Z (ASCII-värde 26) tömmer bildskärmen och placerar markören i övre vänstra hörnet.

CTRL-Ü (ASCII-värde 30) placerar markören i övre vänstra hörnet utan att tömma bildskärmen.

Man kan placera markören i valfri position på bildskärmen genom att skicka ASCII-värde 27,61 följt av två tecken som anger rad och kolumn. Tredje ASCII-tecknet, värde mellan 32 och 55, placerar markören på rad 0 till 23. Fjärde ASCII-tecknet, värde mellan 32 och 71/111, anger kolumn.

Exempel:

Följande ASCII-tecken tas emot: 27,61,42,57. Detta medför att markören placeras på rad 10 och kolumn 25.

För att använda datorn som Datavisionsterminal, måste datorn kopplas om för split speed, se avsnitt 2. (ABC800M går ej att använda som Datavisionsterminal. OptionsPROMmet i ABC806 innehåller ingen drivrutin för Datavision, men drivrutinen kan laddas från skiva, se separat bruksanvisning).

Skriv:

```
LOAD V24:WEA10K24.40A
```

4.2 Terminal

Vill man ha utskrift på fil, t ex skrivare eller flexskiva, skriver man ett V efter enheten V24:. I övrigt byggs parameterblocket upp enligt avsnitt 3. Se programexemplet nedan.

För att kontrollera om man har den nya versionen av options-PROM kan man i ett program skriva:

```
IF PEEK2(PEEK2(65500))<>10 THEN felhanterare
```

För att erhålla en större inputbuffert än 80 tecken skriv:

```
DIM Buffert $\alpha$ =Storlek  
POKE PEEK2(65500)+2,VAROOT(Buffert $\alpha$ ),SWAP%(VAROOT(Buffert $\alpha$ ))  
OPEN "V24:" AS FILE nr
```

De inkommande tecknen lagras i detta fall i strängvariabeln "Buf-
fert".

OBS! POKE måste göras innan man gör OPEN.

Antal tecken i inputbufferten kontrolleras med:

Antal tecken=PEEK2(PEEK2(65500)+6)

För att se hur mycket tecken det finns ledigt i outputbufferten skriver man:

Ledig plats=PEEK2(PEEK2(65500)+4)

Vid användning av GET och PUT har man följande dataformat: 1 startbit, 8 databitar och 1 stoppbit. Med INPUT, INPUT LINE och PRINT är dataformatet följande: 1 startbit, 7 databitar, 1 paritetsbit och 1 stoppbit.

Nedanstående exempel visar hur man kan använda datorn som terminal tillsammans med ett BASIC-program. Ta alltid för vana att avsluta programmen genom att trycka PF1, varvid alla filer stängs. I programexemplen som sänder en fil startas sändningen med PF2.

För ABC802/806 med tangentbord ABC55 finns översättningstabell över PF-tangenterna i respektive bruksanvisning.

```

100 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 REM !
120 REM !      TERMINAL
130 REM !      Exempel på hur man
140 REM !      sparar inkommande
150 REM !      text på fil
160 REM !      Avbryt med PF1
170 REM ! (c)LUXOR DATORER AB  MOTALA
180 REM !      Ver. 1.2
190 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200 EXTEND
210 INTEGER
220 !
230 !
240 !
250 IF PEEK2(PEEK2(65500))<>8 AND PEEK2(PEEK2(65500))<>10 THEN 490
260 DIM Buffer=10000
270 POKE PEEK2(65500)+2,VAROOT(Buffer),SWAP%(VAROOT(Buffer))
280 DIM Filename=16
290 !
300 !
310 !
320 !
330 !
340 PRINT CHR(12) " * Terminal * Spara inkommande tecken "
350 PRINT STRING(PEEK(65364),61)
360 PRINT CUR(3,0) "Sparas under filnamn: ";
370 INPUT "Filename"
380 !
390 !
400 !
410 OPEN "V24:VEA30E24.2" AS FILE 1
420 PREPARE Filename AS FILE 2
430 Dummy=FNPos(1)
440 IF PEEK2(PEEK2(65500)+6)<>0 THEN PRINT FNInchr;
450 IF SYS(5)=128 THEN Dummy=FNInkey
460 IF ASCII(Dummy)=192 THEN 510 ELSE PRINT Dummy;
470 GOTO 430
480 !
490 PRINT "Fel typ av options-PROM!" CHR(7)
500 !
510 CLOSE 1
520 CLOSE 2
530 END
540 !
550 !
560 ! Läs tecken från V24:, skriv på fil
570 !
580 DEF FNInchr LOCAL Dummy=1
590   Dummy=FNPos(0)
600   GET #1,Buffer
610   PUT #2,Buffer
620   Dummy=CHR(ASCII(Buffer) AND 127)
630   RETURN Dummy
640 FNEND
650 !
660 !
670 ! Läs tecken från tangenbord, skriv på V24:
680 !
690 DEF FNInkey LOCAL Dummy=1
700   GET Dummy

```

```

710 IF ASCII(Dummy)=192 THEN RETURN Dummy
720 PRINT #1,Dummy;
730 RETURN Dummy
740 FNEND
750 !
760 !
770 !
780 !
790 DEF FNPos(Släckt) LOCAL Minne
800 IF PEEK(65364)=40 THEN 950
810 !
820 ! Cursor 80 tecken
830 !
840 OUT 56,10
850 OUT 57 40-32*Släckt
860 Minne=30720+PEEK(65363)*80+PEEK(65362)
870 OUT 56,14
880 OUT 57,SWAP%(Minne)
890 OUT 56,15
900 OUT 57,Minne
910 RETURN --
920 !
930 ! Cursor 40 tecken
940 !
950 POKE 31744+PEEK(65363)*128-PEEK(65363)/8*984+PEEK(65362),128*Släckt
960 RETURN --
970 FNEND

```

```

100 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 REM !
120 REM !          TERMINAL          !
130 REM !      Exempel på hur man    !
140 REM !      sänder en binär-fil,  !
145 REM !      t ex ett .BAC-program !
150 REM !      Starta sändning med PF2 !
160 REM !      Avbryt med PF1        !
170 REM ! (c)LUXOR DATORER AB  MOTALA !
180 REM !      Ver. 1.1              !
190 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200 EXTEND
210 INTEGER
220 !
230 !
240 !
250 IF PEEK2(PEEK2(65500))<>8 AND PEEK2(PEEK2(65500))<>10 THEN 500
260 DIM Buffer%=10000
270 POKE PEEK2(65500)+2,VAROOT(Buffer%),SWAP%(VAROOT(Buffer%))
280 DIM Filename%=16
290 !
300 !
310 !
320 !
330 PRINT CHR$(12) " * Terminal * Sänd fil på V24: (CH. B) "
340 PRINT STRING$(PEEK(65364),61)
350 PRINT CUR(3,0) " Ange fil som skall sändas: ";
360 INPUT " "Filename%
370 !
380 !
390 !
400 OPEN "V24:VEA30E24.2" AS FILE 1
410 OPEN Filename% AS FILE 2
420 Dummy%=FNPos%(1)
430 IF Eof=38 OR Start=0 THEN 450
440 IF PEEK2(PEEK2(65500)+4)<>0 THEN Dummy%=FNPutchr%
450 IF PEEK2(PEEK2(65500)+6)<>0 THEN PRINT FNInchr%;
460 IF SYS(5)=128 THEN Dummy%=FNInkey%
470 IF ASCII(Dummy%)=192 THEN 520 ELSE PRINT Dummy%;
480 GOTO 420
490 !
500 PRINT "Fel typ av options-PROM!" CHR$(7)
510 !
520 CLOSE 1
530 END
540 !
550 ! Läs tecken från V24:
560 !
570 DEF FNInchr% LOCAL Dummy%=1
580 Dummy%=FNPos%(0)
590 GET #1,Buffer%
600 Dummy%=CHR$(ASCII(Buffer%) AND 127)
610 RETURN Dummy%
620 FNEND
630 !
640 ! Läs tecken från tangentbord, skriv på V24:
650 !
660 DEF FNInkey% LOCAL Dummy%=1
670 Dummy%=FNPos%(0)
680 GET Dummy%
690 IF ASCII(Dummy%)=193 THEN Start=1 : RETURN " "

```

```

700 IF Start=1 OR ASCII(Dummy)=192 THEN RETURN Dummy
710 PRINT #1,Dummy;
720 RETURN Dummy
730 FNEND
740 !
750 ! Sänd textfil på V24:
760 !
770 DEF FNPutchr LOCAL Buffert=80
780 Dummy=FNPos(0)
790 ON ERROR GOTO 830
800 GET #2,Buffert COUNT PEEK2(PEEK2(65500)+4)-1
810 PUT #1,Buffert
820 RETURN --
830 CLOSE 2
840 Eof=38
850 Start=0
860 RETURN --
870 FNEND
880 !
890 ! Skriv cursor
900 !
910 DEF FNPos(Släckt) LOCAL Minne
920 IF PEEK(65364)=40 THEN 1070
930 !
940 ! Cursor 80 tecken
950 !
960 OUT 56,10
970 OUT 57,40-32*Släckt
980 Minne=30720+PEEK(65363)*80+PEEK(65362)
990 OUT 56,14
1000 OUT 57,SWAP%(Minne)
1010 OUT 56,15
1020 OUT 57,Minne
1030 RETURN --
1040 !
1050 ! Cursor 40 tecken
1060 !
1070 POKE 31744+PEEK(65363)*128-PEEK(65363)/8*984+PEEK(65362),128*Släckt
1080 RETURN --
1090 FNEND

```

```

100 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 REM !
120 REM !      TERMINAL
130 REM !      Exempel på hur man
140 REM !      sänder en textfil
150 REM !      Starta sändning med PF2
160 REM !      Avbryt med PF1
170 REM ! (c)LUXOR DATORER AB MOTALA
180 REM !      Ver. 1.1
190 REM !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200 EXTEND
210 INTEGER
220 !
230 !
240 !
250 IF PEEK2(PEEK2(65500))<>8 AND PEEK2(PEEK2(65500))<>10 THEN 500
260 DIM Buffer=10000
270 POKE PEEK2(65500)+2,VAROOT(Buffer),SWAP%(VAROOT(Buffer))
280 DIM Filename=16
290 DIM Inlin=160
300 !
310 !
320 !
330 PRINT CHR(12) "ASCII-text-filsändning Sänd fil på V24: (CH. B)"
340 PRINT STRING(PEEK(65364),61)
350 PRINT CUR(3,0) "Ange fil som skall sändas:"
360 INPUT "Filename"
370 !
380 !
390 !
400 OPEN "V24:VEA30E24.2" AS FILE 1
410 OPEN Filename AS FILE 2
420 Dummy=FNPos(1)
430 IF Eof=38 OR Start=0 THEN 450
440 IF PEEK2(PEEK2(65500)+4)<>0 THEN Dummy=FNPutchr
450 IF PEEK2(PEEK2(65500)+6)<>0 THEN PRINT FNInchr
460 IF SYS(5)=128 THEN Dummy=FNInkey
470 IF ASCII(Dummy)=192 THEN 520 ELSE PRINT Dummy
480 GOTO 420
490 !
500 PRINT "Fel typ av options-PROM!" CHR(7)
510 !
520 CLOSE 1
530 END
540 !
550 ! Läs tecken från V24:
560 !
570 DEF FNInchr LOCAL Dummy=1
580   Dummy=FNPos(0)
590   GET #1,Buffer
600   Dummy=CHR(ASCII(Buffer) AND 127)
610   RETURN Dummy
620 FNMEND
630 !
640 ! Läs tecken från tangentbord, skriv på V24:
650 !
660 DEF FNInkey LOCAL Dummy=1
670   Dummy=FNPos(0)
680   GET Dummy
690   IF ASCII(Dummy)=193 THEN Start=1 : RETURN
700   IF Start=1 OR ASCII(Dummy)=192 THEN RETURN Dummy

```